

Migration MMapV1 vers Wired Tiger

Sommaire

[Concept](#)
[Préparation](#)
 [Quel format de donnée le MongoDB utilise?](#)
 [Limiter les données sauvegardées en base](#)
 [Faire un état des lieux](#)
[Migration de la base](#)

Préparation

Quel format de donnée votre MongoDB utilise?

Vérifier que le type de moteur de stockage est bien MMapv1 à l'aide de la commande suivante :

```
mongo shinken --quiet --eval "print(db.serverStatus().storageEngine.name)"
```

Le retour de la commande doit être **MMapv1**.



Si le retour est **wiredTiger**, alors la base de données est déjà migrée et vous n'avez pas besoin de continuer cette procédure.

Faire un état des lieux

En premier lieu, il est important de récupérer des statistiques de la base de données qui vont nous servir plupart à vérifier que la migration de donnée s'est bien passé.

```
mongo shinken --quiet --eval 'db.getCollectionNames()' | tee db_info.info  
mongo shinken --quiet --eval 'printjson(db.adminCommand( { listDatabases: 1 } ));' | tee -a db_info.info  
mongo shinken --quiet --eval 'db.getCollection("sla_archive").count();' | tee -a db_info.info
```

Ces commandes vont afficher et conserver dans le fichier db_info.info respectivement :

- La liste des collections de la base *shinken* dans une chaîne de caractères séparée par des virgules.
- La liste des bases présentes sur le serveur de production, dans un format JSON.
- Le nombre d'archives SLA présente sur le serveur de production.

En fonction de votre architecture, il est possible que des commandes ne retournent rien. Si par exemple vous exécutez ces commandes sur un serveur qui héberge uniquement la base du Synchronizer, alors seul la deuxième commande vous retournera des valeurs.



Si aucune commande ne retourne de valeurs, il faut vérifier que vous êtes sur le bon serveur.

Cette procédure changera bien le type du moteur de stockage, mais aucune donnée ne sera impactée.

Enlever ce qui est inutile

Avant de faire la migration, il peut être intéressant de supprimer les anciennes données SLA afin de ne conserver que les données pertinentes.

- Cette étape est **facultative**
- Utiliser le script `shinken-sla-delete-until`
 - Son fonctionnement est décrit dans la page suivante : [shinken-sla-delete-until - Suppression manuelle des archives SLA par jour](#)
 - On peut supprimer les données SLA trop vieilles si l'on n'avait pas mis en place la rotation des SLA.

```
shinken-sla-delete-until --nb-day=365
```

`--nb-day=365` : Pour garder 365 jours de données SLA

- Si vous réduisez le nombre de jours stockés en utilisant le script, changez le paramètre **nb_stored_days** du module sla, par défaut dans /etc/shinken/modules/sla.cfg
 - Cela évitera que de nouveaux jours inutiles soit garder en base.
 - Référez-vous à la page de configuration du module SLA pour plus de d'information (voir [Module SLA](#)).
 - Note : Il n'est nécessaire de redémarrer Shinken maintenant, ce sera fait plus tard dans la procédure.

Migration de la base

Il faut éteindre Shinken pour continuer la procédure

```
service shinken stop
```



Important !

Avant toute opération, faites un shinken-backup complet (ou avec les options qui permettent de sauvegarder les données) du serveur impacté

On sauvegarde les données de Shinken :

```
cd /tmp
mongodump -o data_to_migrate
```

On éteint la base de données et on la vide

```
service mongod stop
cd /var/lib/mongo
rm -fr *
```

Changer le moteur de **mmapv1** à **wiredTiger**. Mettre à jour le fichier */etc/mongod.conf* :

```
storage:
  engine: wiredTiger
```

Démarrer Mongo

```
service mongod start
```

Vérifier le type de moteur de stockage :

```
mongo shinken --quiet --eval "print(db.serverStatus().storageEngine.name)"
```

Le retour de la commande doit être **wiredTiger**

On restaure les données dans MongoDB

```
cd /tmp
mongorestore data_to_migrate
```

On vérifie les données dedans avec le fichier db_info.info que l'on a créé précédemment.



Si vous avez utilisé le script shinken-sla-delete-until, le nombre de sla_archive doit avoir changé.

```
mongo shinken --quiet --eval 'db.getCollectionNames()'  
mongo shinken --quiet --eval 'printjson(db.adminCommand( { listDatabases: 1 } ));'  
mongo shinken --quiet --eval 'db.getCollection("sla_archive").count();'
```

Démarrer Shinken

```
service shinken start
```